

Azure Purview CLI

```
purviewcli
```

Azure Purview

Interface



Purview Studio

[Classifications](#) | [Glossary](#) | [Insights](#) | [Lineage](#) | [Search](#) | [Sensitivity Labels](#) | [Sources](#)



purviewcli

Platform (REST API Endpoints)



Apache Atlas 2.0

catalog

[Entity](#) | [Glossary](#) | [Lineage](#) | [Relationship](#) | [Types](#) | [Search](#)



scan

[Collections](#) | [Sources](#) | [Scans](#) | [Scan Rule Sets](#)



Data Insights

guardian

[Assets](#) | [Scans](#) | [Glossary](#) | [Classification](#)
[Sensitivity Labels](#) | [File Extensions](#)

What is purviewcli?

`purviewcli` is a Python package that provides a cross-platform command line interface (CLI) to Azure Purview. Suitable for ad-hoc tasks and quick exploratory operations.

Installation

```
pip install purviewcli
```

Environment Variables

Environment Variable	Description
PURVIEW_NAME (★)	Azure Purview account name.
AZURE_CLIENT_ID	The Azure AD tenant(directory) ID.
AZURE_TENANT_ID	Client(application) ID of an App Registration.
AZURE_CLIENT_SECRET	Client secret generated for the App Registration.
AZURE_CLIENT_CERTIFICATE_PATH	A path to certificate and private key pair.
AZURE_USERNAME	The username(upn) of an Azure AD user account.
AZURE_PASSWORD	The password of the Azure AD user account.

★ PURVIEW_NAME is mandatory, all other environment variables are optional.

Authentication

The `purviewcli` package leverages `DefaultAzureCredential` from `azure-identity`. `purviewcli` will attempt to authenticate via the following mechanisms in this order, stopping when one succeeds.

Mechanism	Description
Environment	If account information has been specified via Environment Variables, <code>purviewcli</code> will use it to authenticate.
Managed Identity	If deployed to an Azure host with Managed Identity enabled, <code>purviewcli</code> will authenticate with it.
Visual Studio Code	If a user has signed in to the Visual Studio Code Azure Account extension, <code>purviewcli</code> will authenticate as that user.
Azure CLI	If a user has signed in via the Azure CLI <code>az login</code> command, <code>purviewcli</code> will authenticate as that user.

Authorization

The identity executing Azure Purview CLI commands will require the following role assignments:

Role	Description
Purview Data Curator	Has access to the Purview portal and can read all content in Azure Purview except for scan bindings, can edit information about assets, can edit classification definitions and glossary terms, and can apply classifications and glossary terms to assets..
Purview Data Source Administrator	Does not have access to the Purview Portal (the user needs to also be in the Data Reader or Data Curator roles) and can manage all aspects of scanning data into Azure Purview but does not have read or write access to content in Azure Purview beyond those related to scanning.

Commands

Commands	Description
<code>pv entity</code>	Entities are a collection of attributes that model or represent an asset.
<code>pv glossary</code>	A glossary provides vocabulary for business users.
<code>pv lineage</code>	Understand the data's origin and where it moves over time.
<code>pv relationship</code>	Relationships describe connections between two entities.
<code>pv types</code>	A Type is a definition of how a particular object type is stored.
<code>pv scan</code>	Azure Purview scan.
<code>pv insight</code>	Azure Purview insights.
<code>pv search</code>	Azure Purview advanced search.

Usage

```
pv command sub-command --parameter1='value' --parameter2='value'
```

- All parameters are required by default.
- Parameters enclosed with square brackets "[]" are optional.
- Mutually exclusive parameters are enclosed with parens "()" and separated with a pipe "|".
- The "<val>" indicates parameters which require an input (e.g. --parameter=<val>). Input can be specified after a space (e.g. --parameter 'value') or equal "=" sign (e.g. --parameter='value').
- Parameters that do not require an input are boolean operators and are **False** by default and **True** if present (e.g. --ignoreRelationships).
- The ellipsis "..." indicates parameters that are allowed to repeat (e.g. --guid='12345' --guid='23451' --guid='34512')

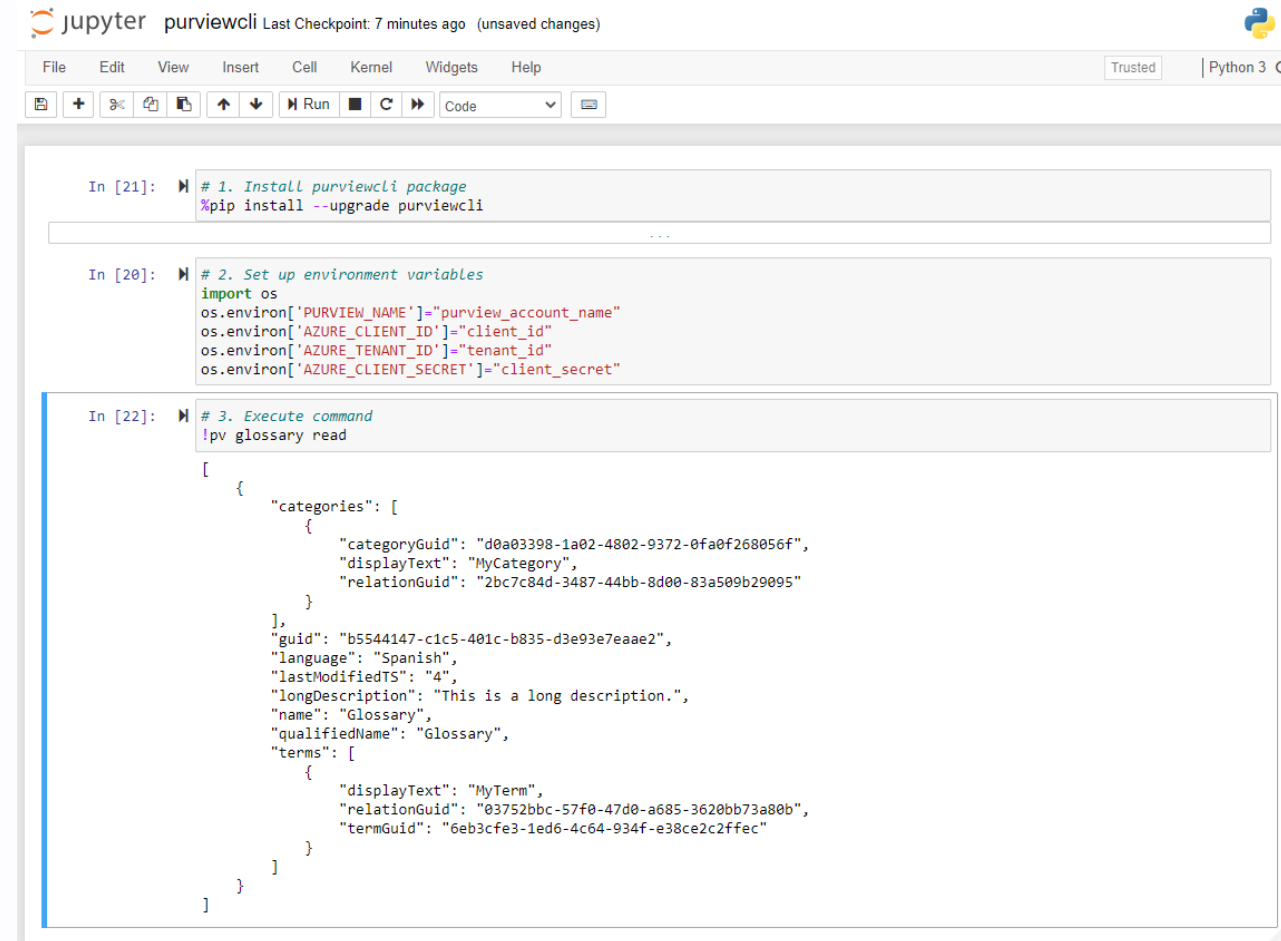
Example

An example of how you can get started with `purviewcli` within a Jupyter Notebook.

1. Install `purviewcli`
2. Set Environment Variables
3. Execute Command

Notebook Samples:

<https://github.com/tayganr/purviewcli/tree/master/samples>



```
Jupyter purviewcli Last Checkpoint: 7 minutes ago (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [21]: # 1. Install purviewcli package
!pip install --upgrade purviewcli

In [20]: # 2. Set up environment variables
import os
os.environ['PURVIEW_NAME'] = "purview_account_name"
os.environ['AZURE_CLIENT_ID'] = "client_id"
os.environ['AZURE_TENANT_ID'] = "tenant_id"
os.environ['AZURE_CLIENT_SECRET'] = "client_secret"

In [22]: # 3. Execute command
!pv glossary read

[
  {
    "categories": [
      {
        "categoryGuid": "d0a03398-1a02-4802-9372-0fa0f268056f",
        "displayText": "MyCategory",
        "relationGuid": "2bc7c84d-3487-44bb-8d00-83a509b29095"
      }
    ],
    "guid": "b5544147-c1c5-401c-b835-d3e93e7eaae2",
    "language": "Spanish",
    "lastModifiedTS": "4",
    "longDescription": "This is a long description.",
    "name": "Glossary",
    "qualifiedName": "Glossary",
    "terms": [
      {
        "displayText": "MyTerm",
        "relationGuid": "03752bbc-57f0-47d0-a685-3620bb73a80b",
        "termGuid": "6eb3cfe3-1ed6-4c64-934f-e38ce2c2ffec"
      }
    ]
  }
]
```

Thank You 🙌